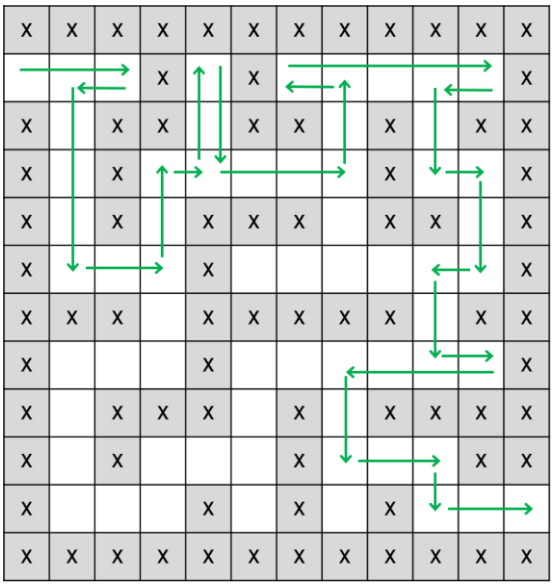
6. Kayla

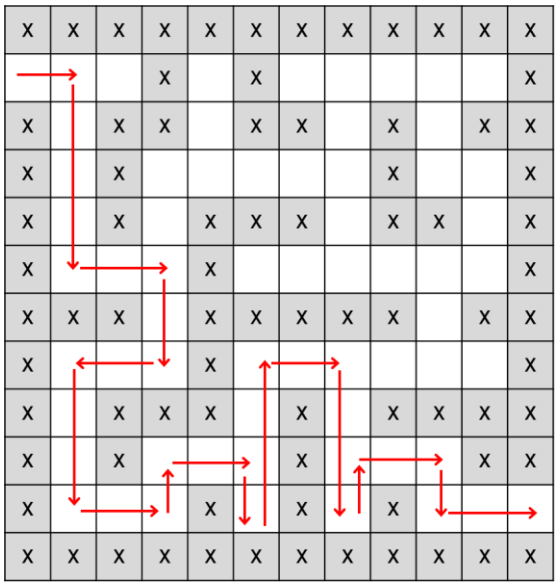
Program Name: Kayla.java **Input File: kayla.dat**

Kayla loves mazes and recently came across this one. Check it out!

You are stepping into a corn maze and need to find the exit. There are two "sure fire" rules to guarantee that you will find the exit – the "left-hand" and "right-hand" rules. Visualize placing your left or right hand on the nearest wall upon entering the maze. You then move through the maze without that hand ever losing contact with the wall until you reach the exit. You will enter dead ends and then turn towards the right for the left-hand rule and toward the left for the right-hand rule. As you step forward with an opening to the left for the left-hand rule, you would keep your hand on the corner of the wall and you would turn to the left and continue. A similar move around a right corner for the right-hand rule allows you to stay in contact with the wall. The process continues until you arrive at the exit. You will never get lost and keep retracing previous paths. However, you may frequently return to a spot you previously visited.

The mazes are the same. The left one uses left-hand rule and right one uses right-hand rule.





There will be exactly one entrance and one exit for each maze and there is always a solution. You may only move horizontally and vertically, no diagonal moves allowed, and the start and finish positions cannot be corners. The top-left corner is always position (1,1).

Input: First line of data file contains the number of test cases, which will not exceed 20. For each test case, the first line contains three items separated by single spaces: the number of rows `R` and columns `C` in the maze, which will not exceed 30, followed by either `'L'` for the left-hand rule or `'R'` for the right-hand rule. The next line contains four integers separated by single spaces: r_1 c_1 r_2 c_2 where (r_1,c_1) is the start position and (r_2,c_2) is the finish position. The start and finish positions will not be adjacent to one another and will not be a corner of the maze. Both `r1` and `r2` will be in the range `1..R` and both `c1` and `c2` will be in the range `1..C`. The next `R` lines will each contain `C` characters separated by single spaces with no extra space at the end of the line. An `'X'` indicates that the position is blocked and cannot be entered while an `'O'` indicates that the position is empty and may be entered.

Output: For each test case, output a list of positions visited in the form "`(r,c)`" where r and `c` are the row and column numbers with no spacing. Display single line containing exactly seven hyphens "`-------`" after each test case.

Kayla (cont)

Sample input:

```
2
12 12 L
2 1 11 12
X X X X X X X X X X X
O O O X O X O O O O O X
X O X X O X X O X O X X
X O X O O O O O X O O X
X O X O X X X O X X O X
X O O O X O O O O O O X
X X X O X X X X X O X X
X O O O X O O O O O O X
X O X X X O X O X X X X
X O X O O O X O O O X X
X O O O X O X O X O O O
X X X X X X X X X X X X
12 12 R
2 1 11 12
X X X X X X X X X X X X
O O O X O X O O O O O X
X O X X O X X O X O X X
X O X O O O O O X O O X
X O X O X X X O X X O X
X O O O X O O O O O O X
X X X O X X X X X O X X
X O O O X O O O O O O X
X O X X X O X O X X X X
X O X O O O X O O O X X
X O O O X O X O X O O O
X X X X X X X X X X X X
```

Sample output:

```
(2,1)
(2,2)
(2,3)
(2,2)
(3,2)
(4,2)
(5,2)
(6,2)
(6,3)
(6,4)
(5,4)
(4,4)
(4,5)
(3,5)
(2,5)
(3,5)
(4,5)
(4,6)
(4,7)
(4,8)
(3,8)
(2,8)
(2,7)
(2,8)
(2,9)
(2,10) – continued at top of next column
```

```
(2,11) – continued from previous column
(2,10)
(3,10)
(4,10)
(4,11)
(5,11)
(6,11)
(6,10)
(7,10)
(8,10)
(8,11)
(8,10)
(8,9)
(8,8)
(9,8)
(10,8)
(10,9)
(10,10)
(11,10)
(11,11)
(11,12)
-------
(2,1)
(2,2)
(3,2)
(4,2)
(5,2)
(6,2)
(6,3)
(6,4)
(7,4)
(8,4)
(8,3)
(8,2)
(9,2)
(10,2)
(11,2)
(11,3)
(11,4)
(10,4)
(10,5)
(10,6)
(11,6)
(10,6)
(9,6)
(8,6)
(8,7)
(8,8)
(9,8)
(10,8)
(11,8)
(10,8)
(10,9)
(10,10)
(11,10)
(11,11)
(11,12)
-------
```